

## KOREAN PATENT ABSTRACTS

(11)Publication  
number:  
(43)Date of publication of application:  
25.05.1999

1019990037501 A

(21)Application number: 1019980045997  
(22)Date of filing: 29.10.1998  
(30)Priority: 30.10.1997 US1997  
960799

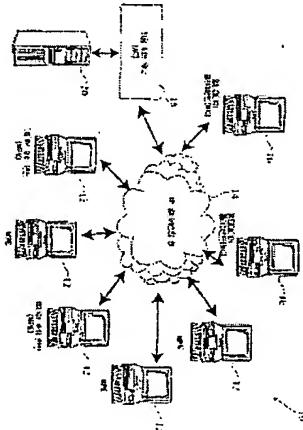
(71)Applicant: MATSUSHITA ELECTRIC  
INDUSTRIAL CO., LTD.  
(72)Inventor: DENNIS BUSHMITCH

(51)Int. CI H04L 12/66

(54) DISTRIBUTED INTERNET PROTOCOL-BASED REAL-TIME MULTIMEDIA STREAMING ARCHITECTURE,  
ESPECIALLY FOR ENABLE COMMUNICATION BETWEEN MULTIPLE MEDIA PUSH ENGINES AND A  
MULTIMEDIA CLIENT

(57) Abstract:

PURPOSE: A distributed internet protocol-based real-time multimedia streaming architecture is provided to readily support data streams of multiple bit rates and perform services at both constant bit rates and variable bit rates. CONSTITUTION: A distributed media delivery system for delivering media selections to a media client over a multicasting network, comprising: a plurality of media push engines(12) accessible through a network; and an admission control system(18) accessible through the network, the admission control system including a catalog for storing the identity of the media selections available for delivery by each of the media push engines.



copyright KIPO 2007

## Legal Status

Date of request for an examination (19991006)

Notification date of refusal decision (00000000)

Final disposal of an application (registration)

Date of final disposal of an application (20011130)

Patent registration number (1003277910000)

Date of registration (20020225)

Number of opposition against the grant of a patent ( )

Date of opposition against the grant of a patent (00000000)

Number of trial against decision to refuse ( )

Date of requesting trial against decision to refuse ( )

Date of extinction of right ( )

**(19) 대한민국특허청(KR)**  
**(12) 공개특허공보(A)**

(51) Int. Cl. 6  
 H04L 12/66

(11) 공개번호 특 1999-0037501  
 (43) 공개일자 1999년 05월 25일

(21) 출원번호 10-1998-0045997  
 (22) 출원일자 1998년 10월 29일

(30) 우선권주장 8/960,799 1997년 10월 30일 미국(US)  
 (71) 출원인 마쓰시타 전기공업주식회사 마사키 우에닷  
 인텔렉츄얼 프라퍼티 센터 카도마 오사카, 571, 일본  
 (72) 발명자 데니스 부쉬미치  
 123 서밋 애버뉴 #2, 서밋, 뉴저지 07901, 미합중국  
 (74) 대리인 김수진  
 윤의섭

심사청구 : 없음

(54) 분배형 인터넷 프로토콜 기반의 실시간 멀티미디어 스트리밍 아키텍처

**요약**

다수의 미디어 푸쉬 엔진이 다중 전달 경로를 가진 멀티캐스팅 망을 통하여 멀티미디어 클라이언트와 통신한다. 전달되는 미디어 선택재(media selection)를 나타내는 스트리밍 데이터(streaming data)가 비계층적 코딩 기법을 사용하여 다수의 미디어 푸쉬 엔진 간에 분배된다. 비계층적 코딩 기법은 데이터가 원래의 데이터 스트림의 모든 컴포넌트의 수보다 적은 수의 컴포넌트로 재구성될 수 있는, 일련의 서브스트림 컴포넌트로 표현된다. 재구성에 사용된 컴포넌트의 수가 많을수록, 재구성된 스트림에 의해 제공되는 서비스의 질은 높아진다. 멀티캐스트 그룹 세션에 대한 허용 제어(admission control)는 분배 방식(distributed fashion)으로 조정되는데, 이에 따르면 허용 제어 유닛이 멀티캐스트 스트림을 개시하고, 모든 후속 허용 제어 결정이 미디어 푸쉬 엔진 스스로에 의해 내려지게 된다. 서브스트림 컴포넌트 데이터는 실시간 전송 프로토콜을 이용하여 보내지는데 반해, 세션 관리 및 분배형 허용 제어 프로세스는 실시간 제어 프로토콜에 의해 조정된다.

**대표도**

도 1

**명세서**

**도면의 간단한 설명**

- 도 1은 본 발명의 바람직한 실시예를 나타내는 망 데이터그램(network datagram)이고.
- 도 2는 두 개의 다른 데이터스트림(data stream)(X와 Y)이 비계층적 다중 기술 코딩(non-hierarchical multiple description coding)을 사용하여 분배되는 방법을 상세하게 나타내는 망 데이터그램이고.
- 도 3은 다중 기술 코딩(MDC)의 하나의 실시예를 나타내는 데이터 흐름도이고.
- 도 4는 TCP/IP 아키텍처를 나타내고, 또한 실시간 프로토콜(Real-time Protocol, RTP)이 IP 기반 시스템에 집적화되는 방법을 나타내는 계층 데이터그램(layer datagram)이고.
- 도 5는 실시간 프로토콜(RTP)에 따른 패킷 포맷(packet format)을 상세하게 나타내는 포맷 데이터그램이고.
- 도 6은 본 발명에 따른 호인가(call admission)와 세션 관리(session management)를 나타내는 망 데이터그램이고.
- 도 7은 멀티캐스팅 IP 망(multicasting IP network)을 통한 미디어 푸쉬 엔진(media push engine, MPE)과 멀티미디어 클라이언트 (multimedia client) 사이의 정보 흐름을 상세하게 나타내는 망 데이터그램이고.

도 8은 소스 컴포넌트 서버의 재분배(source component server redistribution)에 대한 처리를 나타내는 망 데이터그램이고,

도 9는 실시간 프로토콜(RTP)가 신뢰도를 증가시키도록 개선된 방법을 나타내는 프로토콜 데이터그램이다.

\* 도면의 주요부분에 이용된 부호의 설명

- 10: 망 멀티미디어 시스템 12: 미디어 푸쉬 엔진
- 14: 멀티캐스팅 망 16: 멀티미디어 클라이언트
- 18: 허용 제어 유닛 20: 카타로그 서비스 시스템

### 발명의 상세한 설명

#### 발명의 목적

##### 발명이 속하는 기술 및 그 분야 종래기술

본 발명은 망 멀티미디어 시스템에 관한 것으로서, 보다 상세하게는 멀티캐스팅 망(multicasting network)을 이용하여 하나 이상의 미디어 클라이언트에 미디어 선택재(media selection)을 전달하기 위한 미디어 전달 시스템(media delivery system)에 관한 것이다.

인터넷의 폭발적인 성장과 함께 인터넷과 인터넷 프로토콜 기반 망을 사용하여 비디오나 오디오 정보와 같은 멀티미디어 선택재를 전달하는 것에 대한 관심이 고조되고 있다. 대화형 TV(interactive television), 주문형 영화(movie on demand) 및 기타 멀티미디어 푸쉬 기술들이 유망한 애플리케이션으로 꼽히고 있다.

인터넷은 베스트 에포트 전달 서비스, 즉 노력이 많이 드는 전달 서비스(best effort delivery service)를 제공하는 비연결망(connectionless network)이다. 데이터 패킷은 목적하는 수용자(recipient)의 주소를 포함하는 데이터그램의 형태로 전달된다. 망 상의 모든 호스트 단자들은 전달이 완료될 때까지 노드간 데이터그램을 전달할 수 있는 능력을 자체적으로 가지고 있기 때문에 발송자(sender)와 수용자 사이의 특정 연결은 필요하지 않다. 이 데이터그램 패킷 전달 체계는 데이터 패킷의 전달이 보장되지 않는 베스트 에포트 전달 시스템으로 구성된다. 데이터그램 패킷은 전달 가능성을 증가시키기 위해 다른 경로들을 통해 보내진다. 따라서, 망 상에서 하나의 단자가 정체(congestion)를 겪게 되면, 후속 데이터그램들은 그 정체된 단자를 피해 대체 경로를 택하게 된다. 이는 데이터그램 패킷들이 도달하는데 걸리는 시간이 보장되어 있지 않음을 의미한다. 심지어 하나의 메시지에 따른 패킷조차도 잘못되어 전달될 수 있다. 이러한 사실은 어떤 종류의 멀티미디어 데이터가 전달되는 방법에 심각하게 영향을 미친다.

많은 경우에 있어서 멀티미디어 데이터는 실시간 전달을 요구한다. 오디오 또는 비디오 데이터의 경우, 특정 미디어 선택재를 나타내는 데이터 스트림은 적절한 시간적 순서에 따라 전달되어 사용자가 오디오 또는 비디오 선택재를 보내진 상태와 동일한 라이브 상태로 재생할 수 있도록 하는 것이 필요하다. 따라서, 만약 데이터그램 패킷이 다른 전달경로를 택하기 때문에 잘못된 상태로 전달된다면, 사용자의 인터랙티브 TV와 같은 멀티미디어 클라이언트에서의 재생은 뒤죽박죽이 되고 말 것이다.

실시간 프로토콜(RTP)은 현재 인터넷이나 또는 IP 프로토콜 기반의 다른 망을 통하여 실시간으로 내용을 전달하는 것에 대한 사실상의 표준에 해당한다. 실시간 프로토콜은 실시간 애플리케이션이 직접 데이터 전송에 사용할 수 있는 프레임워크를 가지는 통상의 전송 제어 프로토콜(transmission control protocol, TCP)를 대체하고 있다. 현재, RTP 표준은 첫 번째 타입의 메시지, 즉, 미디어 컨텐트 데이터 또는 스트리밍 데이터를 운반하는 것을 지원한다. 전형적으로 별도의 프로토콜인 실시간 제어 프로토콜(Real-Time Control Protocol, RTCP)이 RTP와 함께 사용되어 세션 관리나 비율 조정(rate adaptation) 등과 같은 제어 메시지를 통과시킨다.

실시간 프로토콜이 컴퓨터 망을 통해 멀티미디어 스트리밍 데이터를 전달하는데 사용할 수 있는데 반해, 현재의 아키텍처는 인터넷에 제공되는 것과 같은 베스트 에포트 망 서비스를 사용하여 양질의 프레젠테이션을 제공할 만큼 견고하지 못한 것으로 알려지고 있다. 본 발명은 이 문제를 분배형 미디어 푸쉬 아키텍처를 사용하여 해결하는데, 분배형 미디어 푸쉬 아키텍처는 스트리밍 데이터를 다중 자원(multiple source)으로부터 다중 분배 경로(multiple distribution paths)를 통해 충분히 공급할 수 있다. 이러한 미디어 푸쉬 엔진은 서비스 스트림 컴포넌트(component : 구성 요소)의 비계층적 셋과 같은 스트리밍 데이터를 저장하는 미디어 저장 유닛과 결합되어 있다. 이러한 컴포넌트는 모든 컴포넌트보다 적은 수의 컴포넌트로부터 복원된 스트림으로 대치될 수 있다. 그 결과, 재구성에서 사용된 컴포넌트의 수가 많으면 많을수록 복원된 스트림의 질이 높다.

종래의 시스템들은 몇 개의 컴포넌트를 여타 것들보다 중요한 것으로 처리하는 계층적 코딩 설계를 사용한다. 따라서, 종래의 시스템은 더 중요한 컴포넌트가 항상 전달되는 것이 보증되도록 많은 양의 자원(resource)을 소비하여야 했다. 반대로, 본 발명의 미디어 전달 시스템은 모든 컴포넌트를 동등하게 처리하는 다중 기술 코딩(MDC)과 비계층적 코딩 설계를 사용한다. 따라서, 주어진 서브스트림 컴포넌트가 전달되는 것을 확보하기 위해서 어떠한 특별한 자원의 할당도 필요치 않다. 자연적으로 더 많은 컴포넌트가 전달되면 될수록 더 높은 품질을 이룩할 수 있다. 반면에 종래의 계층적 코딩과는 달리 어떤 소정의 패킷의 하나 손실되더라도 신호의 질은 별로 저하되지 않는다.

분배형 미디어 전달 시스템은 또한 분배 허용 제어 시스템(distributed admission control system)을 채용한다. 미디어 클라이언트는 소정의 미디어 선택재를 요청하도록 하나의 허용 제어 유닛(admission control unit)과 접촉한다. 그러나, 이후 허용 제어 결정은 미디어 푸쉬 엔진 스스로에 의해 분배형 방식으로 처리된다. 허용 제어 유닛은 망 전체에 분배된 다수개의 미디어 푸쉬 엔진과 서로 요청(request)을 주고받고, 상기 푸쉬 엔진은 그들이 멀티캐스팅 세션(session)에 참가할 것인지의 여부를 개별적으로 결정한다. 이런 식으로 개별 미디어 푸쉬 엔진은 국지적 정체 여부를 평가하여 요청된 데이터 스트림을 제공할 수 있는지 여부를 결정한다. 허용 제어 유닛은 어느 미디어 푸쉬 엔진이 멀티캐스트 그룹 세션에 대해 허용되는지를 직접적으로 결정하는데 관여하지 않는다. 허용 제어 유닛은 단지 분배형 구조에서 멀티캐스팅 그룹 세션 어드레스를 할당한 후 허용 프로세스가 분배형 방식을 통해 자발적으로 진행되게 한다.

본 발명의 이해를 돋기 위해서, 그 목적과 효과는 첨부도면 및 명세서를 참조하여 이하 설명한다.

### 발명이 이루고자 하는 기술적 과제

본 발명의 목적은 다중 기술 코딩(multiple description coding, MDC)과 다중 경로 전송(multiple path transport)을 사용하여 정체가 발생할 때, 다중 기술 코딩과 그룹 세션의 참여자들이 추가되거나 그룹으로부터 삭제되는 방법을 통하여 시스템을 중단하지 않고도 전달되는 데이터의 크기를 조절함으로써 비디오 및 오디오 스트리밍 데이터 뿐만 아니라 멀티미디어 선택재의 실시간 전달을 가능하게 하는 분배형 미디어 전달 시스템(distributed media delivery system)을 제공하는 데 있다.

### 발명의 구성 및 작용

도 1을 참조하면, 분배형 망 멀티미디어 시스템이 참조번호(10)으로 예시되어 있다. 다수의 미디어 푸쉬 엔진(12)은 멀티캐스팅 망(14)을 통하여 연결된다. 바람직한 실시예에는 인터넷 프로토콜(IP)을 이용하는 망을 통하여 작동하도록 디자인된다. 그러나, 본 발명의 원리는 다른 프로토콜을 이용한 망에도 쉽게 확장될 수 있다. 망(14)은 도시된 바와 같이 하나 또는 그 이상의 멀티미디어 클라이언트(16)에 의해 억세스된다. 망(14)을 통하여 억세스가 가능한 허용 제어 유닛(18)은 멀티캐스트 그룹 세션을 초기화 또는 개시하기 위해 소정의 허용 제어 기능을 수행한다. 허용 제어 유닛은 카타로그 서비스 시스템(catalog service system)(20)을 포함한다. 이 카타로그 서비스 시스템은 어느 멀티미디어 선택재가 다수개의 미디어 푸쉬 엔진에 의해 전달될 수 있는지를 지시하는 데이터베이스 기록을 포함한다. 비록 멀티캐스팅 그룹 세션을 개시하는 것에 관련된다 할지라도, 이하 상세히 설명되는 것과 같이 허용 제어 과정은 실제 분배형 방식으로 수행된다.

분배형 미디어 전달 시스템은 멀티미디어 클라이언트와, 요청된 전달 가능 미디어 선택재를 갖는 멀티미디어 푸쉬 엔진 사이에서 멀티캐스트 그룹 세션을 개시함으로써 멀티미디어 클라이언트로부터의 전달 요청에 응답한다. 통상 멀티미디어 푸쉬 엔진은 요청된 선택에 대응하는 스트림 데이터를 동시에 전달하는데 관여하고 있다. 멀티미디어 클라이언트는 프레젠테이션 기능을 수행하는 유저 호스트이다. 그것은 관여한 미디어 푸쉬 엔진에 의해 전달된 여러 스트림 컴포넌트로부터 최종 스트림을 재구성한다. 각 미디어 푸쉬 엔진은 스트림 컴포넌트 데이터 스토리지를 갖는다. 그리고, 이러한 데이터 스토리지 시스템은 탑재 가능하고 투명(transparent: 회선의 종별, 전송제어 절차등의 제한을 받지 않는)하며, 회수할 수 있는 기능을 제공하는 적정 분배형 파일시스템에 의해 중재될 수 있다.

분배형 미디어 전달 시스템의 중요한 점은 스트림 데이터가 미디어 푸쉬 엔진에 저장되는 방법이라는 점이다. 비계층적인 구조로 멀티미디어 데이터를 저장하는 종래의 시스템과는 달리, 본 발명은 다중 기술 코딩(MDC)로서 여기에 언급된 것과 같은 비계층적 코딩 설계를 이용한다. 다중 기술 코딩은 비디오 및/ 또는 오디오 스트림을 서브스트림 요구 컴포넌트(substreams called component)로 분산시킨다. 각 컴포넌트는 독립적으로 모든 다른 컴포넌트로부터 망을 통하여 코드화되어 전송된다. 멀티미디어 클라이언트(16)의 클라이언트 소프트웨어는 컴포넌트의 어떠한 서브셋으로부터도 재구성 스트림을 어셈블할 수 있다. 이런 식으로 재구성 스트림은 모든 컴포넌트보다 적은 수의 컴포넌트로부터 어셈블될 수 있으며, 재구성에 사용된 컴포넌트의 숫자가 많으면 많을수록 재구성 스트림의 질은 높아진다.

본질적으로 신뢰할 수 없는 망을 통하여 스트리밍 데이터를 전달하도록 하는 비계층적 코딩의 사용은 특히, 다중 푸쉬 엔진이 전달에 사용될 때 놀랄 만큼 신뢰할 수 있는 미디어 전달을 가능케 한다. 상술하는 바와 같이, 미디어 푸쉬 엔진은 고품질의 서비스를 유지하기 위해 필요한 만큼 미디어 푸쉬 엔진을 그룹 세션에 가감하여 분배형 방식으로 멀티캐스팅 그룹 세션 허용 프로세스 자체를 제어한다. 따라서, 멀티캐스팅 망(14)이 정체를 별로 겪지 않는 경우 다중 기술 코드(MDC)로 인코딩된 스트림의 컴포넌트 모두를 제공하기 위해 단지 적은 수의 미디어 푸쉬 엔진만이 필요할 수 있다. 비록 몇 개의 컴포넌트가 제때 전달되지 않는다 하더라도 멀티미디어 클라이언트는 질적으로 약간 저하된 프레젠테이션용 스트림을 재구성할 수 있다. 만약 망의 정체상태가 심각한 경우, 미디어 푸쉬 엔진은 추가적인 미디어 푸쉬 엔진을 부가하도록 서로서로 조정한다. 허용 제어 과정이 분배형이기 때문에 개별 미디어 푸쉬 엔진은 그들 자신의 국지적(local) 데이터 정체상태를 평가한 후 국지적 통신 상태에 따라 그룹세션에 관여한다.

도 2는 다중 기술 코딩이 어떻게 동작되는지를 상세히 보여주는 도면이다. 도 2에서, 두 개의 멀티미디어 스트림(X, Y)은 다수개의 미디어 푸쉬 엔진의 양단에 저장된다. 이러한 스트림은 서브 스트림 컴포넌트로 나누어지는데 그것은 X1, X2, ..., Xn; Y1, Y2, ..., Yn이다. 다수개의 미디어 푸쉬 엔진 양단에 저장된 서브 스트림이 각 엔진에 대해 동일하지 않다는 것을 주목할 필요가 있다. 따라서, 미디어 푸쉬 엔진(12a)은 컴포넌트(X1, X6 및 Y1)을 저장한다. 동시에 미디어 푸쉬 엔진(12b)은 컴포넌트(X2, X7 및 Y2)를 저장한다.

멀티미디어 클라이언트는 적절한 순서로 적절한 서브스트림 컴포넌트를 합산함으로써 원하는 데이터 스트림을 재어셈블한다. 그렇게 하여 멀티미디어 클라이언트(16a)는 도시된 바와 같이 스트림(X)을 재구성하고, 멀티미디어 클라이언트(16b)는 도시된 바와 같이 스트림(Y)을 구성한다. 멀티미디어 클라이언트에서 개별 서브스트림 컴포넌트는 다른 미디어 푸쉬 엔진으로부터 다른 경로를 통해 도착한다.

바람직한 다중 기술 코딩 설계는 도 3에 도시된 바와 같다. 기본 멀티 미디어 데이터 스트림(예: 비디오 및/또는 오디오 데이터)은 다중 서브신호로 분해되고, 각 서브신호는 독립적으로 압축된다. 전술한 바와 같이, 분해는 비계층적이어서 수신 가능한 신호가 어느 하나의 서브신호로부터 회복되고, 부가 서브신호에 의해 점진적 개선이 실현되며, 모든 서브신호가 정확하게 수신될 때에 완전한 재구성이 달성된다. 더욱이 상기 세 개의 표준조건이 충족되는 한 다시 전체 압축 이득을 최대화하는 것이 바람직하다.

원 신호를 분해하는 한가지 방법은 원 신호에 대한 감소된 해상도 대표값으로서 각각의 서브신호를 재구성하는 것이다. 이것은 원신호를 저역필터를 통하여 처리한 후 다운샘플링(downsampling)을 행함으로써 완료된다. 이렇게 하여 서브신호는 단지 샘플링 위치에 있어서만 차이가 난다. 바람직하게는, 상기 분해는 전치필터(pre-filter) 또는 그 변형(shifted version)을 포함하는 필터뱅크(filter bank)에 의해 얻어질 수 있다.

전치필터는 다운샘플된 서브신호에서 에일리어싱(aliasing) 컴포넌트를 억제하도록 선택되어야 한다. 이것은 서브신호를 코딩하는데 요구되는 비트 비율(bit rate)을 감소시킴과 동시에 하나의 서브신호로부터 이미지 복원을 가능하도록 한다.

전치 필터는 고주파 컴포넌트를 완전히 제거하지 않도록 선택되어야 한다. 고주파 컴포넌트가 완전히 제거되면 모든 서브신호가 존재하더라도 원 신호에 있는 고주파신호를 복원시킬 수 없게 된다. 따라서, 필터는 고주파 컴포넌트를 억제시키되 완전히 제거하지는 않는다.

수학적으로 서브스트림 컴포넌트를 스트림으로 재구성하는 것은 모든 서브스트림에서의 샘플 및 원 스트림에서의 샘플과 연관된 매트릭스 방정식의 역변환과 관련이 있다. 종래에는 이것은 많은 양의 계산 및 메모리 영역을 요하는 역변환(inversion)을 가진, 큰 매트릭스 방정식과 관련된다. 이러한 계산부담을 경감시키기 위한 하나의 방법은 블록 재귀적 재구조 방법(block recursive reconstruction method)을 이용하는 것이다. 상기 재귀적 과정에서의 각각의 단계에서 원 스트림 내에 있는  $2 \times 2$  블록 샘플은 수신된 서브컴포넌트 스트림 내의 네 개까지의 대응 샘플에 기초하여 복원된다. 다른 계산 테크닉을 사용하여 동일한 결과를 얻을 수 있다. 다중 기술 코딩을 이용한 비계층적 구조에서의 코딩 및 앤코딩에 관한 상세한 정보는 "Robust Image Coding and Transport in Wireless Network Using a Non-Hierarchical Decomposition(저자: 정두만, 야오양)", "Mobile Multimedia Communication (Goodman: 플레늄 출판사)" 등을 참고하면 된다.

MDC 코드 서브스트림은 흐름 제어 수행용 RTP 및 내용 전달용 RTCP를 이용하여 데이터 그램의 형태로 멀티미디어 캐스팅 망을 통하여 전달된다. 이러한 프로토콜은 많은 인터넷 애플리케이션에서 사용된 TCP/IP 프로토콜과 당연히 공존할 수 있다. 도 4는 5개의 구성요소(entity)가 어떻게 서로 통신하는가에 대한 일례를 통하여 이러한 프로토콜을 다시 보여준다. 도 5에서 구성요소(30-38)는 인터넷을 통하여 잘 알려진 계층적 아키텍처(architecture)를 이용하여 서로 통신하고 있다. 도 4는 단지 바람직한 실시예로서의 RTP가 하나의 가능한 아키텍

쳐 구조에서 어떻게 통합되는가를 보여주기 위한 것이라는 점에 유의해야 한다. 비록 본 실시예에서는 RTP가 실시되더라도 이는 본 발명의 범주에 한정되는 것이 아니며, 다른 메시지 전달 프로토콜도 적절히 이용될 수 있다.

도 4에서 각 통신 구성요소(30-38)는 도시된 바와 같이 하부에는 물리적 계층이고 상부는 응용계층인 계층적 아키텍처로 되어 있다. 구성요소(30)는 물리적 레벨에서의 이더넷 프로토콜(Ethernet Protocol)을 이용하여 구성요소(32)와 통신한다. 구성요소(32) 및 구성요소(34)는 ATM 프로토콜을 이용하여 물리적 레벨에서 통신한다. 동일한 방식으로 구성요소(34)는 이더넷 프로토콜을 이용하여 구성요소(36)과 통신하고, 구성요소(36)는 PPP 프로토콜을 이용하여 구성요소(38)와 통화한다. 또한, 예시된 물리적 계층 통신 프로토콜은 첨부한 청구범위에서와 같이 본 발명의 권리를 한정하는 것은 아니다.

상기 물리적 계층은 인터넷 프로토콜(IP)층이다. IP 프로토콜은 응용계층으로부터 물리적 또는 트랜스포트 계층을 격리시킨다. IP 프로토콜은 정보 패킷이 데이터그램(datagram)으로 보내지거나 받는 비연결식 통신을 지원한다. 도 4의 보기에서는 모든 통신 구성요소가 IP 프로토콜을 이용하고 있음을 유의해야 한다.

IP 프로토콜 위의 계층은 TCP 프로토콜과 UDP 프로토콜의 2개의 다른 하이레벨 프로토콜로 되어 있다. 이것 또한 가능한 구조의 하나의 예시에 지나지 않는다. UDP 프로토콜 또는 사용자 데이터그램 프로토콜은 단순한 전송 프로토콜을 나타내며, 전달 메시지의 순서의 보존을 꾀하지 않는다. TCP 프로토콜 또는 전달 제어 프로토콜은 높은 신뢰도를 제공하며, 나아가 데이터 그램이 적절한 순서(sequence)로 전달되는 것을 보증한다. TCP 프로토콜은 확인응답 시스템(acknowledge system)을 이용하며, 그 결과 모든 데이터 그램이 적절한 순서로 전달되는 것을 보증한다. TCP 프로토콜은 알려지지 않은 패킷을 재전송하는 매커니즘을 포함한다. 이러한 확인응답 테크닉은 적절한 패킷 전달을 보증하지만, 실시간에 있어서의 패킷전달은 보증할 수 없다. 따라서, TCP 프로토콜은 멀티미디어 비디오 및/또는 오디오 데이터와 같은 실시간 데이터를 전달하는데는 적합하지 않다.

실시간 프로토콜(RTP)은 TCP의 복잡한 전송 프로토콜을 애플리케이션(application)을 직접 사용할 수 있는 단순 체계(simple framework)로 대체한다. 손실 데이터 검색 및 전달지연을 알려주는 재전송 매커니즘을 수행하는 대신에, RTP 프로토콜은 단순히 손실 데이터를 무시한다. RTP 프로토콜은 통상 패킷 전달 순서와 관련되어 있지 않다. 프로토콜은 응용계층이 순서가 뒤바뀐 데이터를 교정한다는 것으로 가정한다. RTP 프로토콜은 MPEG, JPEG 및 H 261과 같은, 많은 다른 인코딩 표준과 호환가능하다.

도 4의 예시에서, 구성요소(30)과 (38)은 런닝 RTP 프로토콜이다. 따라서, 스트리밍 데이터는 구성요소들(30,32,34,36,38)로 구성된 망을 통하여 구성요소(30)로부터 구성요소(38)까지 제공된다.

RTP 프로토콜은 멀티캐스트 동작용으로 디자인된다. 멀티캐스팅은 메시지가 지정된 셋(set)의 많은 다른 수신자에게로 전달되는 메시지 방송의 한 형태이다. 멀티캐스트 어드레스들은 일련의 인터페이스를 확인하는데, 상기 인터페이스는 다른 시스템에 속하는 다중 인터페이스를 자주 포함한다. 메시지가 멀티캐스트지정 어드레스를 가질 때, 망은 셋 내에 있는 모든 인터페이스에게 메시지를 전달하고자 한다. 이러한 기능은 시스템으로 하여금 메시지를 발생하게 하고, 많은 다른 수신자에게 메시지를 전달하게 한다.

데이터그램 패킷을 다중 수신자에게 전달하는 것은 별개로 하고, 멀티캐스팅 망은 통상 메시지 수신자로부터의 피드백을 지원한다. 보통 멀티캐스팅 그룹 세션의 참가자들은 이러한 피드백 메시지를 수신한다. 보통 이러한 피드백 메시지들은 관련된 RTCP에 뒤이어 나오며, 실시간 트래픽 제어용으로 사용된다. 어떤 점에서는 RTCP는 RTP에 대한 선택적인 확장이다. 바람직하게는 RTCP 패킷이 흐름 제어와 그룹 멀티캐스트 세션내에 참가하고 있는 구성요소사이의 세션 메너지먼트 정보를 보내기 위해 사용된다.

도 5는 RTP 패킷 포맷을 보여주고 있다. 패킷은 순번(sequence number)과 적절한 시간 순서대로 패킷을 재어셈블리하는데 사용되는 시간 스템프(time stamp)를 포함하고 있다.

도 6 및 도 7은 멀티미디어 클라이언트, 허용 제어유닛 및 미디어 푸쉬 엔진이 멀티캐스트 그룹 세션 내에서 서로 어떻게 통신을 수행하는가에 대해 상세하게 보여주고 있다. 특히, 도 6은 바람직한 실시예에서의 통신 순서 및 기본 메시지 흐름을 보여주고 있다. 도 7은 RTP 및 RTCP 프로토콜이 미디어 푸쉬 엔진으로부터 멀티미디어 클라이언트에게 서브스트림 컴포넌트 데이터그램을 전송시키는데 어떻게 이용되는지에 대한 상세도를 보여주고 있다.

도 6을 참조하면, 멀티미디어 클라이언트(16)는 유니캐스트(unicast) TCP 프로토콜 메시지를 허용 제어 유닛(18)으로 보내고, 특정 미디어 선택재의 전달 개시를 요청한다. 허용 제어 유닛(18)은 카타로그 서비스 시스템(20)과 협의하여 요청된 선택재(요청스트림)가 망 상에 존재하는지를 결정한다. 스트림이 존재하면, 허용 제어 유닛은 스트림 개시 메시지를 적어도 몇 개의 요청된 선택재의 서브스트림 컴포넌트를 갖는 미디어 푸쉬 엔진(12)으로 전송

한다. 이러한 개시 스트림 요청은 모든 미디어 푸쉬 엔진으로 보내어진다. 요청된 스트림 컴포넌트를 제공할 수 있다고 판단한 미디어 푸쉬 엔진은 특정 스트림을 제공 및 수신하는 호스트(host) 사이의 멀티캐스팅 세션 관리 및 흐름 제어 세션에 참여한다.

참여한 미디어 푸쉬 엔진과 멀티미디어 클라이언트는 허용 제어 유닛으로부터 그러한 제어 멀티캐스트 그룹 세션 용에 대하여 필요한 멀티캐스트 어드레스를 확보한다. 이후 허용 제어 유닛은 세션으로부터 효과적으로 이탈되고, 서브시퀀스 세션 관리 및 흐름 제어 메시지가 멀티캐스트 그룹 멤버(즉, 멀티미디어 클라이언트와 모든 해당 미디어 푸쉬 엔진) 사이에서만 교환되도록 한다. 이것은 허용 제어 유닛(18)상에 있는 오버헤드(overhead)를 감소 시킨다.

허용 제어 유닛은 멀티캐스트 세션에 의해 사용되는 멀티캐스트 분류 D 어드레스를 발생시킨다. 이러한 어드레스는 매우 적은 가용성 멀티캐스트 어드레스 엔트리의 풀(pool)로부터 선택된다. 이에 따라 허용 제어 유닛은 멀티캐스트 어드레스의 할당을 관리하게 된다. 멀티캐스트 세션이 종료되면, 허용 제어 유닛은 가용어드레스의 풀로 멀티캐스트 세션 어드레스를 되돌려 보낸다.

일단 멀티캐스트 그룹 세션이 시작되면, 서브스트림 컴포넌트를 제공하는 미디어 푸쉬 엔진들은 멀티미디어 클라이언트(16)으로 유니캐스트 RTP 세션 스트림 데이터를 보냄으로써 서브스트림 컴포넌트를 제공한다. RTCP 프로토콜을 통하여 이러한 미디어 푸쉬 엔진은 서로 통신을 하며, 고품질의 서비스를 유지하는데 필요한 만큼 멀티캐스트 세션을 합치거나 분리시킨다.

도 7을 참조하면, 미디어 푸쉬 엔진과 멀티미디어 클라이언트는 두 개의 다른 레벨에서 망을 통하여 통신한다. 점선으로 나타낸 바와 같이, 유니캐스트 RTP 세션은 멀티미디어 스트리밍 데이터를 멀티미디어 클라이언트에 전송한다. 동시에, 요구된 바와 같이 미디어 푸쉬 엔진과 멀티미디어 클라이언트는 제어 명령과 다른 세션 관리 명령(예를 들어, Start Push, Pause, Continue) 외에도 각기 다른 RTCP 기록, 좀더 구체적으로는 송신 기록(sender report)과 수신 기록(received report)을 보낸다. RTCP 제어 신호는 도 7에서 양방향 실선 화살표로 표시되어 있다.

본질적으로 스트림 개시 메시지(Stream Open Message)가 허용 제어 유닛에 의해 전송된 후, 각 미디어 푸쉬 엔진은 연관된 미디어 스토리지 시스템과 협의하여 요청된 스트림 컴포넌트를 제공하는 것이 가능한지 여부를 결정한다. 만약 가능하다면, 미디어 푸쉬 엔진은 특정의 멀티캐스트 그룹에 참여한다. 그렇지 않고 또한 이후 별도의 요청이 없다면 멀티캐스트 세션에 참가하지 않는다. 일단 미디어 푸쉬 엔진이 멀티캐스트 그룹에 참여하였다면, 그것은 RTCP 프로토콜을 사용하여 통신에 참여하며, 송신된 데이터와 수신된 데이터에 대한 통계치는 RTCP 프로토콜에 따라 그룹 멤버들 간에 교환된다. 전술한 바와 같이, 허용 제어 유닛은 이러한 통신에 참여할 필요가 없고, 따라서 또 다른 세션에 대한 요청이 없거나 현재 세션에 대한 종료 요청이 있을 때까지 허용 제어 유닛은 휴지 상태에 머무른다.

효과적으로는 시스템은 분배형 허용 제어 시스템의 역할을 수행하는데, 이 시스템에서는 그룹의 참여자들이 공동으로(collectively) 및 배분하여(distributively) 허용 제어 결정을 내린다. 이러한 분배형 접근법의 장점은 발명이 가능하고, 메커니즘(intelligent mechanism)과 통합됨으로써 멀티캐스트 망이 실시간 전달을 보장하지 못하는 베스트 에포트 망이라는 사실에도 불구하고 망 정체를 방지하고 서비스의 질을 향상시킬 수 있다는 점이다.

베스트 에포트 망, 특히 복잡한 통신상태 및 사용자 제어 체계가 부족한 망들은 자주 정체상태를 경험한다. 그러한 정체상태는 실시간 데이터의 손실 또는 심각한 지연을 초래한다. 전술한 바와 같이, 늦게 전달된 데이터는 실질적으로 전달되지 않은 것으로 처리된다. 정체상태의 망 단자로 데이터가 지속적으로 입력되는 것은 정체상태를 더욱 악화시키는 경향이 있다. 본 발명은 정체상태가 감지되었을 때 주어진 노드가 전송 중인 컴포넌트의 수를 줄일 필요가 있다는 표시로 감시성(感時性, time-sensitive) 전송에 대한 RTCP 송신 기록과 수신 기록을 사용한다.

도 4는 이것이 어떻게 실행되는지를 보여준다. 멀티미디어 클라이언트(16)는 서브스트림 컴포넌트 X1, X2, X3, X4로 구성된 실시간 데이터 스트림 X를 요청하였다. 도 7의 미디어 푸쉬 엔진(12)이 국지적 정체상태에 있어서 서브스트림 컴포넌트가 멀티미디어 클라이언트(16)에 늦게 도달한다고 가정한다. 멀티미디어 클라이언트의 RTCP 프로토콜 수신 기록은 미디어 푸쉬 엔진(12)으로부터의 일정 부분의 컴포넌트 데이터에 대해 미디어 푸쉬 엔진(12)(그룹 세션에 참여한 모든 다른 미디어 푸쉬 엔진을 포함하여)에 알려준다. 미디어 푸쉬 엔진(12)은 이러한 기록들을 분석하고 선택된 컴포넌트, 이 경우 X3의 송신을 중단함으로써 정체지점을 통과하는 통신량을 감소시킨다. 따라서, 전술한 조정(adjustment) 후에는 미디어 푸쉬 엔진(12)은 단지 X1, X2, X4의 컴포넌트만을 멀티미디어 클라이언트로 공급한다. 그룹 세션에 참여한 다른 미디어 푸쉬 엔진들이 동일한 송신 및 수신 기록들을 받기 때문에 만약 다른 푸쉬 엔진들이 이 손실 컴포넌트 X3를 제공할 수 있다면 미디어 푸쉬 엔진(12)으로부터의 X3 컴포넌트의 손실은 보상될 수 있다. 그렇지 않다면 서비스의 질은 전술한 바와 같이 약간 저하될 것이다.

도 8은 전송된 서브스트림 컴포넌트들에 대한 국지적인 조정을 행함으로써 데이터 스트림이 어떻게 효과적으로 재분배되는지를 보여준다. 예시된 실시예에서 미디어 푸쉬 엔진(12b)으로부터의 서브스트림 컴포넌트를 공급하는 데이터 경로 중의 어떤가에 국지적인 정체가 있다고 가정한다. RTCP 송신 및 수신 기록들은 미디어 푸쉬 엔진(12b)에 의해 멀티미디어 클라이언트(16)에 보내진 컴포넌트 중 일부분이 손실 또는 국지적 정체로 인해 지연되고 있음을 알려준다. 본 실시예에서 손실된 컴포넌트는 미디어 푸쉬 엔진(12a)의 스토리지 시스템에도 존재하고 있다. 미디어 푸쉬 엔진(12a)은 멀티미디어 클라이언트에 손실된 컴포넌트 페이로드를 재전송하거나 단지 장래의 실시간 데이터 처리시 전송될 컴포넌트의 셋을 조정한다. 손실된 페이로드가 다른 미디어 푸쉬 엔진에 의해 재전송되는 경우, 데이터 스트림이 재구성되어 사용자에게 보내지기 전에 손실된 컴포넌트가 이미 전달된 컴포넌트와 재어셈블될 수 있도록 충분한 버퍼가 멀티미디어 클라이언트에 제공되어야 한다. 시스템이 단지 장래의 전송을 위해 컴포넌트 셋을 변경하는 경우라면, 측정 가능한 서버 컴포넌트 재분배 메커니즘이 요구된다. 이 메커니즘은 멀티미디어 클라이언트의 스트림 데이터의 제공을 향상시킴으로써 서비스의 질의 향상을 도모한다.

전술한 실시예가 일반적으로 대부분의 미디어 전달 애플리케이션에 적당하지만, 때로는 약간의 질적 저하도 인정되지 않는 시스템이 있다. 그러한 시스템은 고화질 방송 영상 분배 시스템을 포함한다. 이러한 보다 까다로운 시스템에서는 전술한 실시예의 시스템은 실시간 컴포넌트에 대한 부가적인 신뢰 메커니즘(reliability mechanism)을 채용할 수 있도록 변경된다. 이 경우 실시간 프로토콜(RTP)은 손실된 실시간 페이로드를 재전송할 수 있도록 변경 또는 증가된다. 이 "신뢰할 수 있는 RTP(Reliability RTP)"는 도 9에 도시되어 있다. 미디어 푸쉬 엔진은 RTP를 사용하여 실시간 스택과 통신한다. 이 경우 제 1 및 제 3의 컴포넌트는 수신되나 제 2의 컴포넌트는 손실된다고 가정한다. 이때 미디어 푸쉬 엔진이 제 2의 페이로드를 받지 못했다는 내용의 즉각적인 접수 부인(negative acknowledgement, NACK)이 RTP 스택으로부터 있게 된다. 다음 미디어 푸쉬 엔진은 필요한 페이로드를 재전송하고 RTP 스택은 필요로 하는 페이로드를 데이터 버퍼 내의 올바른 위치에 위치시키게 된다. 다음으로 클라이언트 애플리케이션은 데이터 버퍼로부터 데이터를 읽어 들이게 된다. 어떤 중복된 패킷이나 과도하게 지연된 패킷도 제거된다.

### 발명의 효과

전술한 바로부터 본 발명은 베스트 에포트 망을 통하여 스트리밍 데이터를 전달하기 위한 분배형 망 기술을 채용하는 미디어 전달 시스템 아키텍처를 제공한다. 상기 아키텍처는 크거나 작거나 간에 서버의 복잡성이 클라이언트의 수와 선형적으로 비례하여 증가할 때 평가가 용이하다. 또한, 상기 아키텍처는 완전분배형이며 단단히 결합된 병렬 아키텍처로서 단순하지만 신뢰성 있는 서비스를 제공한다.

다중 기술 코딩(MDC)과 다중 경로 전송(multiple path transport)을 사용하여, 본 발명은 지연 발생시의 전송 재시도 기술(delay-producing transmission retry technique)에 의존하지 않고 고품질의 서비스를 제공할 수 있다. 따라서, 본 발명은 현재의 데이터 전송용 실시간 전송 프로토콜(RTP)과 세션 관리, 비율 조정 등을 위한 실시간 제어 프로토콜(RTCP)과 함께 용이하게 동작할 수 있다. 정체가 발생할 때, 다중 기술 코딩과 그룹 세션의 참여자들이 추가되거나 그룹으로부터 삭제되는 방법을 통하여 시스템을 중단하지 않고도 전달되는 데이터의 크기를 조절할 수 있다. 스트림의 흐름 제어 역시 동일한 메커니즘을 사용하여 제어할 수 있어서 정체 상태가 감지되면 이를 방지 또는 감소시킬 수 있다.

따라서, 본 발명은 비디오 및 오디오 스트리밍 데이터 뿐만 아니라 멀티미디어 선택재의 전달에 이상적으로 적합하다. 본 발명은 쉽게 다중 비트 레이트의 데이터 스트림을 지원하며, 고정 비트 레이트나 가변 비트 레이트로 서비스를 제공할 수도 있다.

본 발명은 바람직한 실시예에 의해 설명되었지만, 첨부된 특허청구범위에 개시된 바와 같이 본 발명의 본질을 변경하지 않는 범위 내에서 어떠한 수정이나 변경도 본 발명의 권리범위에 속한다.

### (57) 청구의 범위

#### 청구항 1

멀티캐스팅 망(multicasting network)을 통하여 미디어 클라이언트에 미디어 선택재(media selection)를 전달하기 위한 분배형 미디어 전달 시스템(distributed media delivery system)에 있어서,

상기 망을 통하여 접속 가능한 다수의 미디어 푸쉬 엔진(media push engine):

모든 컴포넌트보다 적은 수의 컴포넌트로부터 복원된 스트림(reconstructed stream)으로 재구성될 수 있는 서브 스트림 컴포넌트의 비계층적 셋(non-hierarchical set)의 형태로 스트리밍 데이터(streaming data)를 저장하도록 되어 있는 미디어 스토리지 유닛(media storage unit): 그리고,

상기 망을 통하여 접속가능하며, 상기 미디어 푸쉬 엔진 각각에 의해 전달되는 미디어 선택재의 아이덴티티(identity)를 저장하기 위한 카탈로그(catalog)를 포함하는 허용 제어 시스템(admission control system)을 구비하며.

상기 스트리밍 데이터는 전달가능한 상기 미디어 선택재를 나타내고, 상기 미디어 푸쉬 엔진은 각각 상기 스트리밍 데이터를 저장하기 위한, 관련된 상기 미디어 스토리지 유닛을 구비하며.

상기 허용 제어 시스템은 미디어 클라이언트로부터의 소정의 미디어 선택재에 대한 요청에 응답하여 상기 미디어 클라이언트 간의 멀티캐스트 그룹 세션을 개시하도록 동작하고, 상기 미디어 푸쉬 엔진의 적어도 일부는 전달가능한 상기 소정의 미디어 선택재를 구비함으로써.

상기 미디어 클라이언트에 상기 소정의 미디어 선택재를 전달하거나 상기 미디어 클라이언트에 의해 상기 소정의 미디어 선택재를 재구성할 수 있도록 멀티캐스트 그룹 세션에 참가한 상기 미디어 푸쉬 엔진이 각각 상기 소정의 미디어 선택재에 해당하는 서브스트림 컴포넌트를 상기 망에 공급하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항2

제 1항에 있어서,

상기 허용 제어 시스템은 적어도 일부분이 상기 미디어 푸쉬 엔진 간의 상호작용에 의해 정의되는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항3

제 1항에 있어서,

상기 미디어 푸쉬 엔진이 다른 통신 경로(communication path)를 통하여 상기 망과 통신하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항4

제 1항에 있어서,

상기 망은 베스트 에포트 전달 서비스(best effort delivery service)를 제공하는 비연결망(connectionless network)인 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항5

제 1항에 있어서,

상기 망은 인터넷(Internet)인 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항6

제 1항에 있어서,

상기 멀티캐스트 그룹 세션에 참가한 상기 미디어 클라이언트와 상기 미디어 푸쉬 엔진은 데이터 전송용 실시간 전송 프로토콜(Real-Time Transport Protocol, RTP)을 채용하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항7

제 1항에 있어서,

상기 멀티캐스트 그룹 세션에 참가한 상기 미디어 클라이언트와 상기 미디어 푸쉬 엔진은 세션 관리(session management)용 실시간 제어 프로토콜(Real-Time Control Protocol, RTCP)을 채용하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항8

제 1항에 있어서,

적어도 상기 서브스트림 컴포넌트의 일부가 다수의 미디어 푸쉬 엔진 간에 반복되는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항9

제 1항에 있어서,

적어도 상기 서브스트림 컴포넌트의 일부가 제 1 및 제 2의 미디어 푸쉬 엔진 간에 반복되고, 상기 전달 시스템이 상기 멀티캐스트 그룹 세션에 참여한 상기 제 1 및 제 2의 미디어 푸쉬 엔진 중 하나를 정체(congestion)의 원인으로 확인하고 자동적으로 상기 제 1 및 제 2의 미디어 푸쉬 엔진 중 다른 하나를 호출하는 정체 통제 시스템(congestion handling system)을 더욱더 구비하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항10

제 9항에 있어서,

상기 분배형 미디어 전달 시스템이 상기 미디어 클라이언트와 결합되어 있고, 서브스트림 컴포넌트를 재구성 전에 저장하기 위한 데이터 버퍼 시스템(data buffering system)을 더욱더 구비하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항11

제 1항에 있어서,

상기 허용 제어 유닛은 멀티캐스트 그룹 세션을 종료하라는 상기 미디어 클라이언트로부터의 요청(request)에 응답하여 상기 멀티캐스트 그룹 세션에 참가한 모든 미디어 푸쉬 엔진에게 상기 세션을 종료할 것을 지시하도록 동작하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항12

제 1항에 있어서,

상기 허용 제어 시스템은 멀티캐스트 그룹 세션을 호출할 때 사용하기 위한 멀티캐스트 세션 어드레스의 풀(pool)을 유지하는 허용 제어 유닛을 포함하고, 상기 허용 제어 유닛은 상기 멀티캐스트 그룹 세션에 의해 사용될 수 있도록 상기 풀로부터 선택된 소정의 멀티캐스트 세션 어드레스를 할당하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항13

제 12항에 있어서,

상기 허용 제어 유닛은 멀티캐스트 그룹 세션을 종료하라는 상기 미디어 클라이언트로부터의 요청에 응답하여 상기 소정의 멀티캐스트 세션 어드레스를 상기 풀에 되돌리도록 동작하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

#### 청구항14

제 12항에 있어서,

상기 멀티캐스트 그룹 세션에 참여한 상기 미디어 클라이언트와 상기 미디어 푸쉬 엔진이 상기 허용 제어 유닛의 관여없이 흐름 제어 메시지(flow control message)를 교환하는 것을 특징으로 하는 분배형 미디어 전달 시스템.

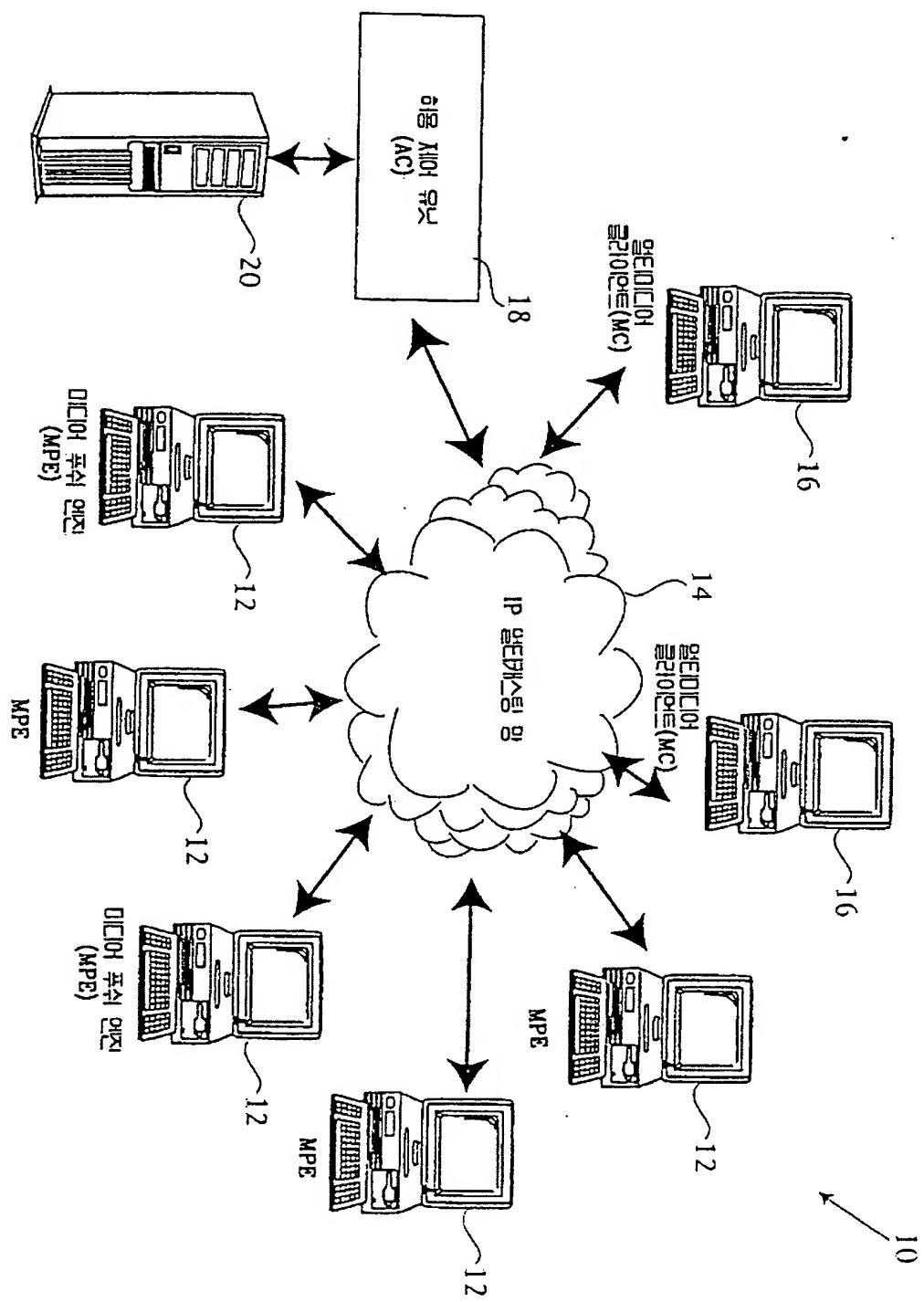
#### 청구항15

제 1항에 있어서,

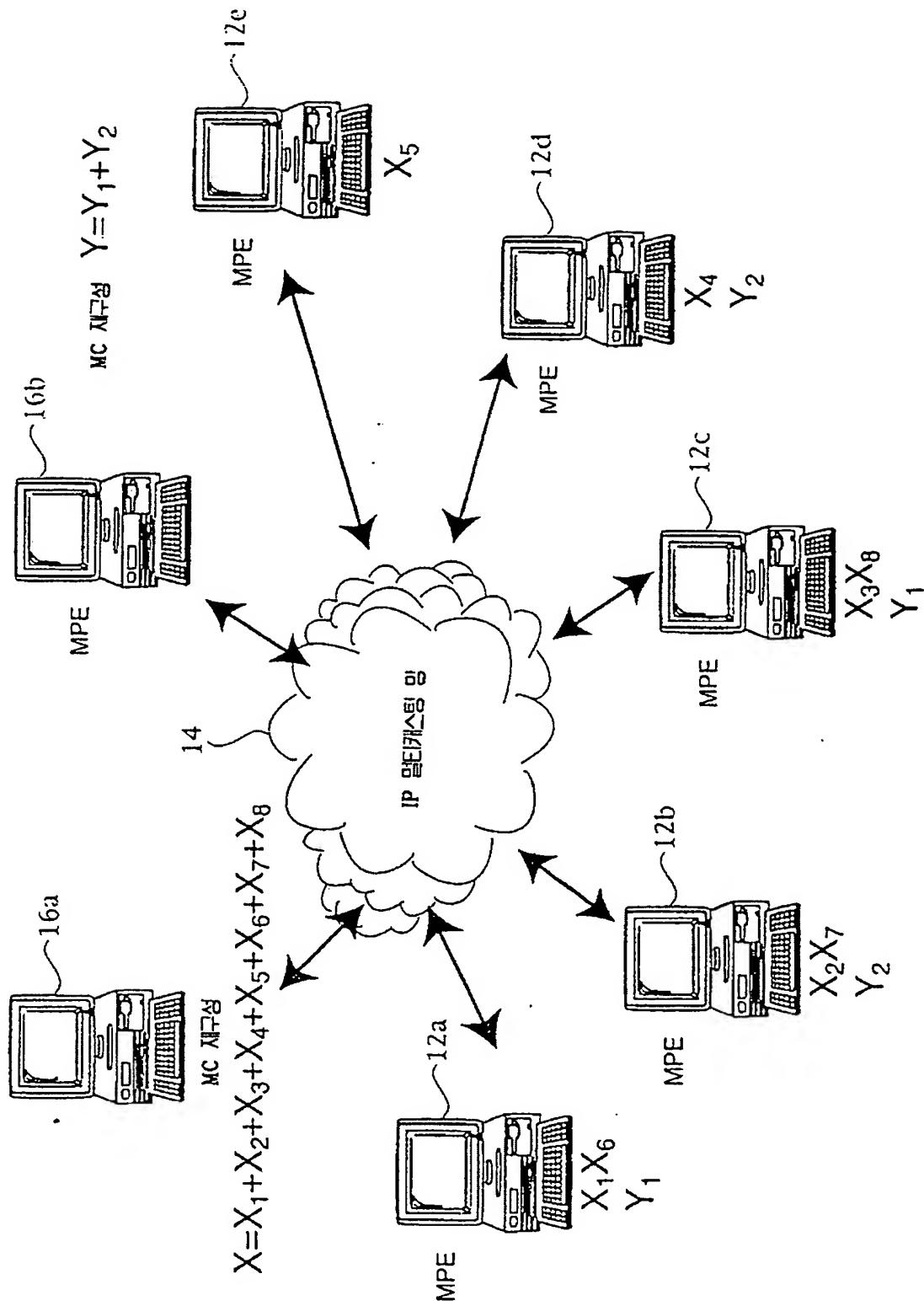
상기 멀티캐스트 그룹 세션에 참여한 상기 미디어 클라이언트와 모든 미디어 푸쉬 엔진 사이에 실시간 스트림 컴포넌트 데이터(real-time stream component data)를 포함하는 데이터 그램의 유니캐스트 흐름(unicast flow)이 있는 것을 특징으로 하는 분배형 미디어 전달 시스템.

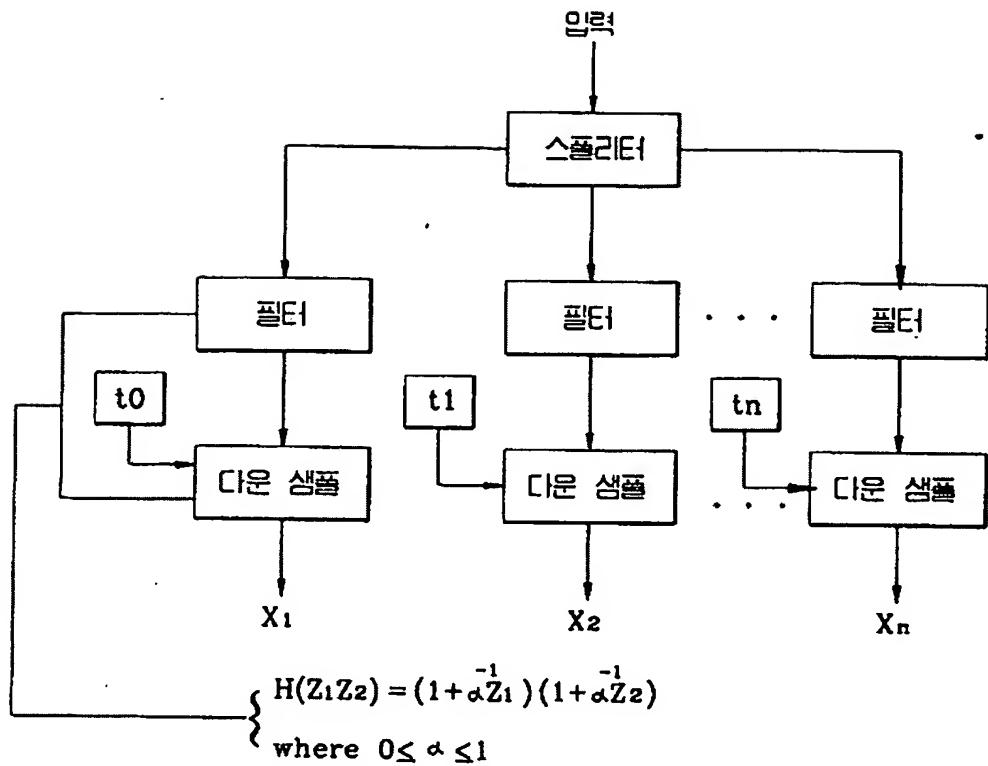
### 도면

#### 도면1



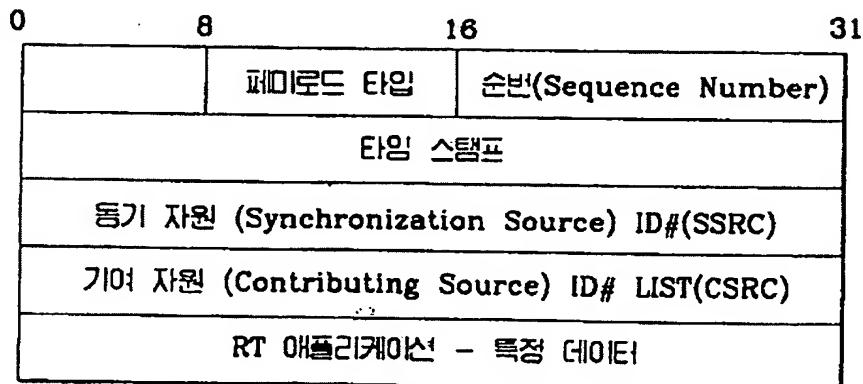
도면2





도면5

### RTP 패킷 포맷



도면4

